A Direct Method for Implicit Particle-In-Cell Simulation

We describe a new method for solving the set of coupled particle and field equations arising in implicit formulations of particle-in-cell plasma simulation. Such implicit integration schemes are necessary for the efficient study of low frequency, long wavelength plasma phenomena; in particular, it is important to be able to select a time-step which is larger than the plasma period when plasma oscillations are not themselves of interest. In addition, it is desirable to employ expressions for time derivatives which affort selective damping of high-frequency modes while preserving low-frequency behavior. Unlike other methods described recently, the method we describe does not involve the introduction of moment equations to advance field quantities in time. The essence of our method is the linearization of the charge density at the advanced time about an approximate density which does not involve the field at the advanced time; this involves the accumulation of coefficients which are introduced into the elliptic field equation to include the effect of the advanced field. Some illustrative results are presented.

I. INTRODUCTION

We have made several advances in the quest for more efficient simulation of low-frequency plasma phenomena. The most adaptable and reliable tools for study of kinetic plasma behavior are the "particle" codes, but the stability of these codes has previously required resolution of the electron plasma period in the time integration, even when the phenomenon under study took place on the much-longer ion time scale.

Analogous limitations on time step in other problems, such as heat flow and chemical rate equations, are overcome through the use of *implicit* time integration schemes. In particle codes, although implicit methods have been analyzed theoretically,[1] their application has been inhibited by the very large number of nonlinear equations to be solved simultaneously, about equal to the number of zone quantities (electric and magnetic fields) plus the much larger number of particle coordinates.

We have begun to experiment with a new method for solution of these equations in two steps. In this method, equations are first set up for the fields. Although these equations involve information from the particles, the number of equations to solve simultaneously is only the number of field

Comments Plasma Phys. Controlled Fusion
1981, Vol. 6, No. 6 pp. 225–236
0374-2806/81/0606–0225/$06.50/0

Printed in the U.S.A.

quantities, defined on the zones. The resulting matrix equations are sparse and their solution is convenient with known methods. Once the fields are known, the particle coordinates can be readily solved for serially, one particle at a time. Here, we will discuss simulations having only the electrostatic field.

First we consider finite-differenced equations of motion for the particles which have the necessary stability at large time-step and are accurate for the low frequency phenomena to be studied. In Ref. 3, a centered expression for the time derivative is employed. Use of this expression leads to a numerically stable algorithm at large $\omega_p \Delta t$, but by itself does not damp high frequency components of the motion, which are aliased into the Nyquist frequency—stable odd-even oscillations of large amplitude can persist (as will be shown in the first example below). Here, ω_p is the plasma frequency and Δt is the time step. In Ref. 2, the derivatives are biased using a scheme to advance positions x and velocities v such as

$$v_{n+1} - v_n = \left(\frac{3}{4}a_{n+1} + \frac{1}{4}a_{n-1}\right)\Delta t, \tag{1a}$$

$$x_{n+1} - x_n = \left(\frac{3}{4}v_{n+1} + \frac{1}{4}v_{n-1}\right)\Delta t, \tag{1b}$$

where subscripts denote time levels. This scheme damps unwanted high-frequency oscillations, while low frequencies $\omega \ll \Delta t^{-1}$ are very weakly damped, as desired: $(\mathrm{Im}\omega)/\omega = \mathcal{O}(\omega\Delta t)^3$. Such schemes are members of a class whose application to plasma simulation has been analyzed in detail in Ref. 1. We have also devised a different class of schemes, whose simplest member is

$$2(v_{n+1/2} - v_{n-1/2}) - (v_{n-1/2} - v_{n-3/2}) = a_{n+1}\Delta t, \tag{2a}$$

$$x_{n+1} - x_n = v_{n+1/2}\Delta t. \tag{2b}$$

This scheme has the same order of accuracy as Eq. (1), while requiring less storage of time levels. The presence of the acceleration at only the $n + 1$ time level increases the damping of high-frequency oscillations. The optimum design of these difference equations is the first issue in practical implementation of large time-step methods.

In all implicit schemes the new positions x_{n+1} depend on the accelerations a_{n+1} due to the electric field E_{n+1}. But this field is not yet known, as it depends on the density ρ_{n+1} of particle positions $\{x_{n+1}\}$. The solution of these coupled particle and field equations is the other major implementation issue.

5037-34

In the method developed by Denavit[2] and Mason[3] for this solution, the fields at the new time level are predicted by solving coupled field and fluid equations, in which the kinetic stress tensor is approximately evaluated from particle velocities known at the earlier time. After the fields are known, the particles are advanced to the new time-level, and, if desired, an improved kinetic stress tensor is calculated and the process iterated.

It is also practical to predict the future electric field E_{n+1} quite directly by means of a linearization of the particle-field equations. This approach may offer greater economy in storage and complexity than methods which require the accumulation of higher velocity moments and the solution of fluid equations for each species. Since this method and its implementation in the experimental code BAAL have not been described previously, we outline the concept here.

Section II below describes a conceptual (gridless) form of the algorithm we have developed. Section III briefly describes the cloud-in-cell algorithm used in our testbed code (BAAL). Section IV presents some illustrative results, and Section V summarizes this work and presents some possible avenues for futher study. Further details of this work, more sophisticated implicit PIC methods, and our analyses of time-differencing algorithms will be presented in future publications.

II. CONCEPTUAL ALGORITHM

The position x_{n+1} of a particle at time level t_{n+1}, as given by an implicit time integration scheme, can be written as

$$x_{n+1} = \beta \Delta t^2 a_{n+1} + x_{n+1}^{(0)}, \tag{3}$$

where $0 < \beta \lesssim 1$ and $x_{n+1}^{(0)}$ is the position obtained from the equation of motion with the acceleration a_{n+1} neglected. Since $x_{n+1}^{(0)}$ depends only on positions and accelerations at times t_n and earlier, it is known. In its simplest form, the BAAL algorithm is derived by linearization of the particle positions relative to $x_{n+1}^{(0)}$.

One can regard the actual position, x_{n+1}, as $x_{n+1}^{(0)}$ plus a displacement $\delta x = \beta \Delta t^2 a_{n+1}$. We form a charge density $\rho_{n+1}^{(0)}$ from $\{x_{n+1}^{(0)}\}$; the actual charge distribution is then $\rho_{n+1}^{(0)}$, plus the change $\delta\rho$ brought about by displacing particles by the amount $\delta x = x_{n+1} - x_{n+1}^{(0)}$. Linearized, this increment to ρ is[1]

$$\delta\rho = -\nabla \cdot [\rho_{n+1}^{(0)}(x)\delta x(x)]. \tag{4}$$

To the same order of approximation, the displacement $\delta x(x)$ of all particles with $x_{n+1}^{(0)} \cong x$ is obtained with a_{n+1} evaluated at x, i.e.,

$$\delta x(x) \cong \beta \Delta t^2 (q/m) E_{n+1}(x). \tag{5}$$

We then have

$$\delta \rho(x) = -\nabla \cdot [\chi(x) E_{n+1}(x)], \tag{6}$$

where the effective susceptibility is

$$\chi(x) = \beta [\rho_{n+1}^{(0)}(x) q/m] \Delta t^2 = \beta \omega_p^2(x) \Delta t^2 \tag{7}$$

(rationalized cgs units are employed). Note that χ depends only on the particle positions $\{x_{n+1}^{(0)}\}$, and not at all on velocity information as required in the moment equation methods.

With these two source contributions, the Poisson equation becomes

$$\nabla \cdot E_{n+1} = \rho_{n+1}^{(0)} - \nabla \cdot (\chi E_{n+1}) \tag{8}$$

or

$$-\nabla \cdot [1 + \chi] \nabla \varphi_{n+1} = \rho_{n+1}^{(0)}. \tag{9}$$

This elliptic equation is solved by standard methods. The field $-\nabla \varphi_{n+1}$ and (5) are then used to calculate the positions $\{x_{n+1}\}$. This algorithm is reminiscent of the method for solution for the vector potential A in some magneto-inductive plasma simulation codes.[4,5]

Should it be necessary, we have shown how to refine the approximations used above by: linearization about a more accurate prediction of x_{n+1} than $x_{n+1}^{(0)}$, iteration, and a more accurate evaluation of δx.

In our spatial-difference representation of these equations, described in the following section, Eq. (4) becomes the gradient of a zonal ρ with respect to particle position, and δx in Eq. (5) depends on the electric field in two zones, using the usual interpolation. In this way we are assured that the density ρ_{n+1} of final particle positions $\{x_{n+1}\}$ satisfies the code's representation of the Poisson equation, $-\nabla^2 \varphi_{n+1} = \rho_{n+1}$, so that desirable features built into the time-differencing scheme will be realized in practice.

III. CLOUD-IN-CELL ALGORITHM

The testbed computer simulation program we have developed, BAAL, employs well-known particle-in-cell techniques to assure a smooth interpolation of the zonal electric field onto the particle locations and of the particle charge onto the charge-density grid. For simplicity we illustrate the scheme

with a fully implicit (backward differenced) model algorithm; the actual code is far more general.

In the model algorithm, each particle k is advanced via

$$v_k^{n+1} = v_k^n + (q/m)\Delta t E^{n+1}(x_k^{n+1}), \tag{10a}$$

$$x_k^{n+1} = x_k^n + \Delta t v_k^{n+1} \tag{10b}$$

(time levels will henceforth be denoted by superscripts, particle and cell indices by subscripts). For each cell j, we write the charge density as

$$\rho_j^{n+1} = (q/\Delta x)\sum_k S(x_k^{n+1} - x_j), \tag{11}$$

where Δx is the cell size and the "shape function" S is the finite-size particle analogue of the Dirac delta function. The calculation of E^{n+1} requires knowledge of ρ^{n+1} before we have explicit knowledge of the x_k^{n+1}. In essence, the BAAL algorithm calculates this charge density implicitly by approximating it as a linear function of E^{n+1}. We compute a simple approximation $\bar{x}_k$ (this is $x_{n+1}^{(0)}$ of the previous section) to x_k^{n+1} via

$$\bar{x}_k = x_k^n + \Delta t v_k^n, \tag{12}$$

which for this simple model is just a "free-streaming" location. We expand the shape function in a Taylor series:

$$S(x_k^{n+1} - x_j) \approx S(\bar{x}_k - x_j) + (x_k^{n+1} - \bar{x}_k)\partial S(\bar{x}_k - x_j)/\partial \bar{x}_k, \tag{13}$$

and note that the expansion is exact for linear-spline particles which do not cross cell boundaries as a result of the new acceleration, i.e., which occupy the same cells at x_k^{n+1} and $\bar{x}_k$. We approximate the coefficient of the first-derivative term via

$$x_k^{n+1} - \bar{x}_k = (q/m)\Delta t^2 E^{n+1}(x_k^{n+1}) \approx (q/m)\Delta t^2 \sum_i S(\bar{x}_k - x_i)E_i^{n+1}, \tag{14}$$

with a relative error of order $\omega_t^2 \Delta t^2 = qE\Delta t^2/mL_E \ll 1$, where ω_t is the particle trapping or bounce frequency and L_E is a typical scale length over which E varies. This approximation imposes a limitation (shared by the moment-equation method) on L_E and $|E|$ for the simulation to be valid.

Using Eqs. (11), (13), and (14), the field equation solved is

$$-(\varphi_{j-1}^{n+1} - 2\varphi_j^{n+1} + \varphi_{j+1}^{n+1})/\Delta x^2 = \rho_{j\,(\mathrm{conv})} + \sum_i W_{ij}E_i, \tag{15}$$

where the "conventional" charge density is

$$\rho_{j\,(\mathrm{conv})} = \sum_k (q/\Delta x)S(\bar{x}_k - x_j), \tag{16}$$

and the "weights" W_{ij} couple each cell to its neighbors when particles are present:

$$W_{ij} = (q^2\Delta t^2/m\Delta x)\sum_k S(\tilde{x}_k - x_i)\partial S(\tilde{x}_k - x_j)/\partial\tilde{x}_k. \tag{17}$$

For linear splines $W_{ij} = 0$ whenever $|i-j| > 1$ so the matrix equation remains sparse. Since for linear splines $\partial S/\partial\tilde{x}_k = \pm 1/\Delta x$ (or zero), there is considerable redundancy in the calculation of the $\rho_{j\,(\mathrm{conv})}$ and the W_{ij}, and it turns out that (at least for the unmagnetized 1d electrostatic code) the number of operations needed to weight all necessary quantities to the grid is the same as the number needed in the conventional electrostatic code. Since the field interpolation and charge deposition steps tend to dominate the particle processing time on a vector computer like the Cray-1, the algorithm is potentially very efficient.

The field equation is written in terms of the electrostatic potential φ, and yields a pentadiagonal linear system plus corner elements arising from the periodic boundary conditions. This system is solved by direct Gaussian inversion, which is straightforward in one dimension but nontrivial to generalize to higher dimensional or electromagnetic problems. We have also solved the field equations by successive overrelaxation, with observed convergence ranging from rapid to very slow depending upon the problem (the diagonal dominance of the matrix is sometimes inadequate). An *ad hoc* spatial filtering is provided for through use of a fast Fourier transform applied either to ρ and χ (i.e., W) before the field solution or to φ afterward; unlike the explicit case, it is not equivalent to filter the input or output of the field equation, since the effect of the spatially varying χ is to couple modes of various wavelengths. We are experimenting with means of applying spatial filtering within the field equation itself.

IV. CODE RESULTS

As a first check of code performance, we initialized a cold electron plasma (with immobile ions) in a run with parameters $\omega_p = 1.0$, $\Delta t = 10.0$ (run "CL27"). For this run, the BAAL testbed used a time-centered trapezoidal rule (centered implicit) version of the model difference scheme described in this section. Thirty-two cells, and 128 particles, were employed, and there was an initial excitation of mode 1 (the longest wavelength mode). The energy in this mode, when plotted as a function of time, exhibits an undamped oscillation with a period of 160. More detailed diagnostics show that the mode is in fact nearly an odd–even oscillation in time, i.e., with a frequency differing from the Nyquist frequency $\pi/\Delta t$ by only the small

shift $\delta\omega = 2\pi/160$. To understand this, we observe that the dispersion relation for this trapezoidal rule scheme is $2/\omega_p\Delta t = \cot \omega\Delta t/2 = \tan(\pi/2 - \omega\Delta t/2)$, yielding a mode frequency $\omega \approx \pi/\Delta t - 4/\omega_p\Delta t^2$, so that the frequency shift is $\delta\omega = 4/\omega_p\Delta t^2 = 0.04$ leading to a period of 157. Similar behavior was observed for $\Delta t = 25.0$.

When the difference scheme of Eq. (1) is employed, the odd–even oscillations (aliased plasma wave) are observed to damp out rapidly, with $\gamma_{obs} \approx -0.07$. This compares well with the predicted damping factor $\gamma_{pred} = -0.069$. Similar damping is obtained when "stiffly stable"[6] schemes such as that of Eq. (2) are employed. This test problem serves to verify that the implicit equations are solved correctly.

As a first model problem, we consider the electron–electron two-stream instability arising from the relative motion of two cold beams of infinite spatial extent. The first run made, "EE13," was a benchmark, using the conventional electrostatic leapfrog simulation algorithm and a small time-step, $\Delta t = 0.25$ with $\omega_p(\text{total}) = 1.0$. For this run, a grid of 32 cells was employed, and 128 particles were used to represent each beam. A spatial filter retaining only the longest-wavelength field mode was employed. Figure 1 is a plot of the "thermal" (kinetic-drift) energy of one beam as a function of time for this run; the analytically predicted growth rate for the parameters chosen, normalized to the plasma frequency, is $\gamma_{pred} = 0.0196$, with a corresponding ten-folding time for field quantities of 117.5. For interpretation of the figure, this number must be divided by 2 since energies exponentiate at twice the rate of field quantities, and so finally we expect the thermal energy to grow by a decade each 58.75 time units. As can be seen in the figure, the observed decade time is quite close to this predicted value.

For comparison with the above run, another ("EE14") was made using the BAAL testbed code. This time-step, $\Delta t = 2.5$, was ten times as large as that of the benchmark, and was large enough that had it been used in an explicit code a numerical instability would have resulted. The time-centered trapezoidal rule version of the model algorithm of this section was employed. Thus, there was no damping of the aliased plasma oscillations at the Nyquist frequency (these would not be visible in a field energy plot anyhow, since only the square of the electric field enters into that diagnostic). The spatial Fourier filtering in this run was applied to ρ and χ. The beam thermal energy history of this run is shown in Figure 2. The instability is visible at an earlier time in the BAAL simulation; presumably this is due to a larger initial level of excitation of the unstable mode. Once visible growth begins, it is clear that the growth rates in this run and the previous run are very nearly equal; while the saturation levels also agree closely, this is not a strong check since by the end of the run almost all of the directed

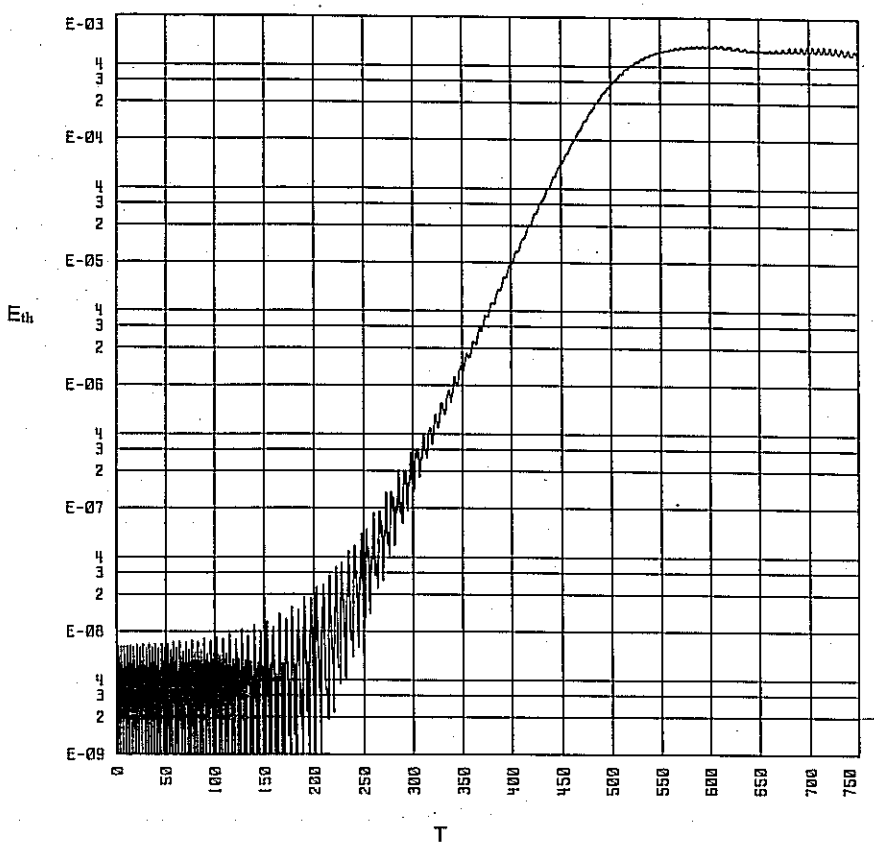

FIGURE 1 Thermal energy versus time for benchmark e–e two-stream run EE13.

kinetic energy of the beams has been converted to thermal energy. For a quantitative check of the growth rate, Figure 3 shows the history of the modulus-squared of the lowest spatial mode density component of one beam. From this figure, the decade time can be measured to be 58.3 ± 0.5 time units, in excellent agreement with the predicted value. Similar behavior of this instability has been observed using implicit moment methods.[2]

As a second model problem, we consider an ion-acoustic traveling wave, with $\omega_{pe} = 1.0$, $\Delta t = 3.0$ (run "IA05"). For this run, the time-level biasing of Eq. (1) was employed, as was a spatial filtering of φ which discarded all but modes 1–4, and which filtered modes 3 and 4 rather heavily. Mode 2 was excited, there were 128 grid cells, the mass ratio m_i/m_e was 16, and there were 2048 electrons and 512 ions. A "quiet start" initial loading was employed. The system length L was 16π, the electron thermal velocity $v_{th,e}$

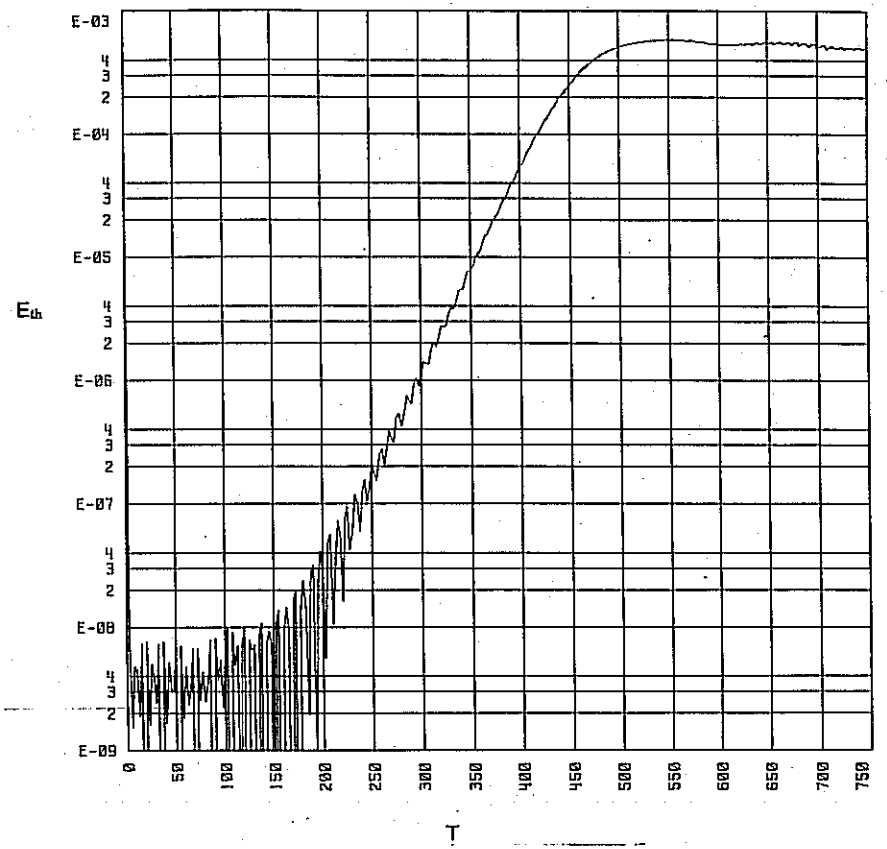

FIGURE 2 Thermal energy versus time for BAAL two-stream run EE14.

was 0.25 (the ions were cold), the Debye length λ_D was 0.25, and the wave vector k was 0.25. The sound speed c_s was 0.0625, and the wave frequency ω was 0.0156 leading to a period τ of 402. The smoothed ion density as a function of x at the beginning of the run, and that at time $t = 210$ (slightly more than half a wave period later), exhibit sinusoidal variation, the latter being of smaller amplitude. One predicts that the left-hand peak should have moved $210c_s = 13.1$ units during this period of 70 time-steps, and measures a traveled distance of 13.2, slightly more than half a wavelength. Thus, the phase velocity of the wave is essentially correct. The decay in amplitude during this interval can be attributed to electron Landau damping, with a rate γ given by [7]

$$|\gamma/\omega_r| = \{\pi m_e/[8m_i(1 + k^2\lambda_D^2)^3]\}^{1/2}, \tag{18}$$

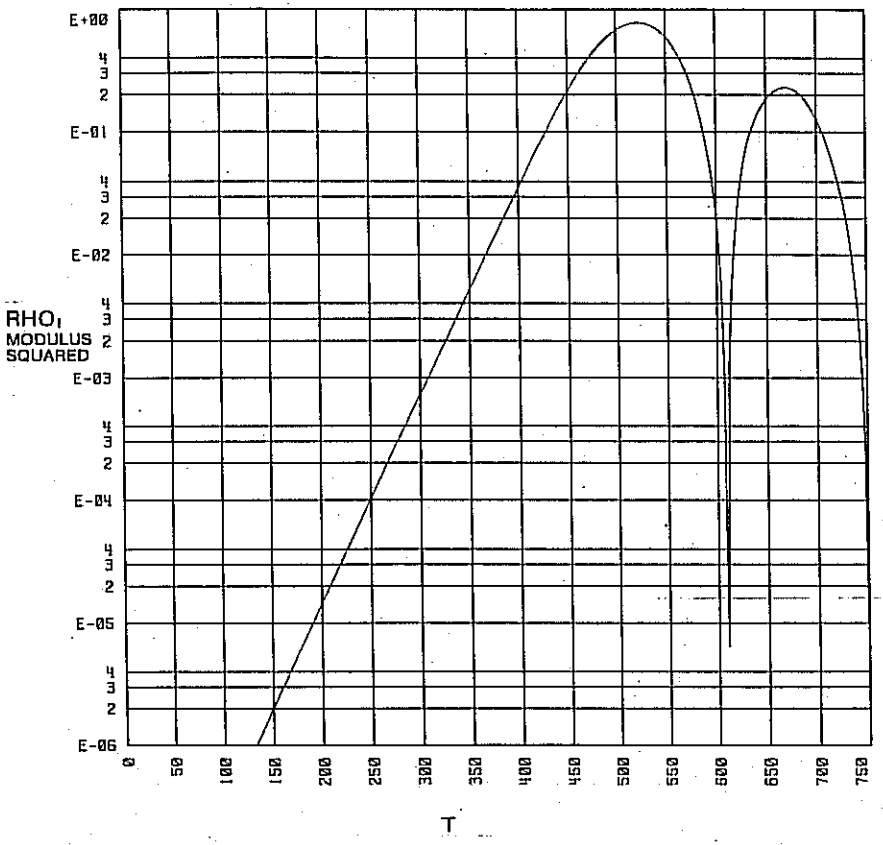

FIGURE 3 Lowest spatial mode density component for one beam versus time for BAAL e–e two-stream run EE14.

which yields $\gamma = -0.00243$, and thus an amplitude $|E_{t=210}| = \exp(-210\gamma) \times |E_{t=0}| \approx (0.6)(0.025) = 0.0150$, quite close to the observed value of 0.0158. That the agreement is this good is largely fortuitous; a history plot of the modulus-squared of the ion charge density reveals a damping of this mode with a rate difficult to measure but in rough ($\pm 20\%$) agreement with the predicted value, out to times of order 350, the falloff being slower at later times. The additional damping associated with the time integration scheme is negligible for this low frequency mode. At times after 250 or so, mode 4 comes up out of the noise and the clean sinusoidal structure observed in the earlier snapshots disappears. The total energy is observed to diminish by about 1% over the course of the run (out to time 600). The bulk of this loss appears in the electron thermal energy, indicating that the particle-pushing scheme induces a small amount of anomalous cooling.

5037-42

V. SUMMARY AND PLANS FOR FURTHER STUDY

We have described a new method for solving the coupled particle and field equations arising in an implicit particle-in-cell formulation of plasma simulation. To date, the direct-implicit PIC approach has been verified in application to ion-acoustic oscillation and two-stream instability, with $\omega_p \Delta t$ in the range 2.5–4.0. Numerical stability and correct dispersion of the testbed code have been verified for a cold simulation plasma up to $\omega_p \Delta t = 25$.

We are delineating the limitations of implicit methods and learning how to surmount them. For example, it is observed that an unphysical cooling of the electrons occurs when uncentered equations of motion are used. This side effect is related to the damping of *low* frequency oscillations; both are smaller in our schemes described above which have third-order damping, than in lower-order algorithms.[3] Our analysis[8] provides guidance to design algorithms which retain desirable dissipation of high-frequency oscillations while minimizing these undesirable effects.

Our future plans include the modeling of warm two-stream instabilities wherein the physically fastest-growing mode is well resolved in the simulation, and the verification of the correct behavior of a plasma expanding into vacuum. We are considering simplifications to the algorithm in order to facilitate extension to 2d and/or electromagnetic codes. Practical application of these methods to problems in inertial and magnetic confinement fusion might then be made.

Acknowledgments

It is a pleasure to acknowledge valuable conversations with J. Denavit, R. J. Mason, D. Potter, V. Thomas, and D. S. Kershaw. This collaboration was stimulated in part by a workshop conducted in 1980 at U. C. Berkeley organized by Professor C. K. Birdsall, who has encouraged our investigation of long-time-step methods in plasma simulation. This work was performed under the auspices of the U.S. Department of Energy, by the Lawrence Livermore Laboratory under Contract No. W-7405-Eng-48. During the early stages of this work, one of us (A.F.) was supported at the U. C. Berkeley Electronics Research Laboratory under contract No. DOE-AS03-76SF00034-DE-AT03-76ET53064.

ALEX FRIEDMAN, A. BRUCE LANGDON, and BRUCE I. COHEN
Lawrence Livermore National Laboratory, University of California, Livermore, California 94550

Contributed by W. Kruer

(Received June 29, 1981)

References

1. A. Bruce Langdon, J. Comput. Phys. **30**, 202 (1979).
2. J. Denavit, "Time Filtering Particle Simulations with $\omega_{pe}\Delta t \gg 1$," UCRL-85097, to appear in J. Comput. Phys.
3. R. J. Mason, "Implicit Moment Particle Simulation of Plasmas," to appear in J. Comput. Phys.
4. A. Friedman, R. L. Ferch, R. N. Sudan and A. T. Drobot, Plasma Phys. **19**, 1101 (1977); A. Mankofsky, A. Friedman and R. N. Sudan, Plasma Phys. **23**, 521(1981).
5. D. O. Dickman, R. L. Morse and C. W. Nielson, Phys. Fluids **12**, 1708 (1969).
6. C. F. Curtiss and J. O. Hirschfelder, Proc. Nat. Acad. Sci. U. S. **38**, 235 (1952).
7. N. A. Krall and A. W. Trivelpiece, *Principles of Plasma Physics* (McGraw-Hill, New York, 1973), p. 390.
8. A. B. Langdon, A. Friedman, and B. I. Cohen, in preparation. B. I. Cohen, A. B. Langdon, and A. Friedman, in preparation.

5037-44